

Claims

1. A method for updating a seed file to match a target file, said method comprising:
generating target file checking data for one or more blocks of said target file;
storing at least a portion of said target file checking data in a cache;
5 receiving seed file checking data corresponding to one or more blocks of said
seed file;
comparing said seed file checking data with said target file checking data to
identify differences in blocks of said seed file and blocks of said target file; and
transmitting information for revising seed file blocks which are different from
10 target file blocks such that said seed file blocks match said target file blocks.
2. The method of claim 1, wherein said target file checking data and said seed file
checking data each comprise weak level checking data and strong level checking data,
and wherein said comparing comprises comparing said weak level checking data and next
comparing strong level checking data only if a match is identified in said weak level
15 checking data.
3. The method of claim 1, wherein said target file checking data and said seed file
checking data each comprise a 32-bit checksum and a 128-bit checksum.
4. The method of claim 1, wherein said target file checking data and said seed file
checking data each comprise weak level checking data and strong level checking data,
20 and wherein said storing comprises storing said weak level checking data associated with
said target file and storing only said strong level checking data associated with said target
file expected to match strong level checking data associated with said seed file.

5. The method of claim 1, wherein said target file checking data and said seed file checking data each comprise a checksum.

6. The method of claim 1, wherein said target file checking data stored in a cache are used with multiple updating requests received from a plurality of clients.

5 7. The method of claim 1, further comprising decompressing said target file prior to said generating.

8. The method of claim 1, wherein said seed file and said target file are decompressed prior to said generating, wherein said seed file blocks are revised in accordance with said transmitted information to match said target file blocks, and
10 wherein said revised seed file blocks are recompressed after revising.

9. The method of claim 8, wherein said seed file comprises a compressed payload, previously separated from a compound file, and wherein said revised seed file is appended to a header file after said recompressing to constitute a revised compound file.

10. The method of claim 9, wherein said compound file and said revised compound
15 file comport with an RPM Package Manager format.

11. A method for updating a seed file to match a target file, said method comprising:
generating seed file checking data for one or more blocks of said seed file;
transmitting said seed file checking data for comparison against cached target file checking data corresponding to one or more blocks of said target file to identify
20 differences in blocks of said seed file and blocks of said target file; and
receiving information for revising seed file blocks which are different from target file blocks such that said seed file blocks match said target file blocks.

12. The method of claim 11, further comprising:

decompressing said seed file prior to said generating;

revising said seed file blocks in accordance with said information to match said target file blocks; and

recompressing said revised seed file blocks.

5 13. The method of claim 12, wherein said seed file comprises a compressed payload, previously separated from a compound file, and wherein said revised seed file blocks are appended to a header file after said recompressing to constitute a revised compound file.

14. The method of claim 13, wherein said compound file and said revised compound file comport with an RPM Package Manager format.

10 15. A computer program product, residing on a computer-readable medium, for use in updating a seed file to match a target file, said computer program product comprising instructions for causing a computer to:

generate target file checking data for one or more blocks of said target file;

store at least a portion of said target file checking data in a cache;

15 receive seed file checking data corresponding to one or more blocks of said seed file;

compare said seed file checking data with said target file checking data to identify differences in blocks of said seed file and blocks of said target file; and

20 transmit information for revising seed file blocks which are different from target file blocks such that said seed file blocks match said target file blocks.

16. The computer program product of claim 15, wherein said target file checking data and said seed file checking data each comprise weak level checking data and strong level checking data, and wherein said computer program product further comprises instructions

for causing said computer to compare said weak level checking data and to compare said strong level checking data only if a match is identified in said weak level checking data.

17. The computer program product of claim 15, wherein said target file checking data and said seed file checking data each comprise weak level checking data and strong level checking data, and wherein said computer program product further comprises instructions
5 for causing said computer to store said weak level checking data associated with said target file and to store only said strong level checking data associated with said target file expected to match strong level checking data associated with said seed file.

18. The computer program product of claim 15, wherein said target file checking data
10 and said seed file checking data each comprise a checksum.

19. A computer program product, residing on a computer-readable medium, for use in updating a seed file to match a target file, said computer program product comprising instructions for causing a computer to:

generate seed file checking data for one or more blocks of said seed file;
15 transmit said seed file checking data for comparison against cached target file checking data corresponding to one or more blocks of said target file to identify differences in blocks of said seed file and blocks of said target file; and

receive information for revising seed file blocks which are different from target file blocks such that said seed file blocks match said target file blocks.

20. A system for updating a seed file to match a target file, said system comprising:
means for generating target file checking data for one or more blocks of said target file;

means for storing at least a portion of said target file checking data in a cache;

means for receiving seed file checking data corresponding to one or more blocks of said seed file;

means for comparing said seed file checking data with said target file checking data to identify differences in blocks of said seed file and blocks of said target file; and

5 means for transmitting information for revising seed file blocks which are different from target file blocks such that said seed file blocks match said target file blocks.